AD-A133 763 AN OVERVIEW OF THE QUASI-OPTIMIZER SYSTEM(U) ARIZONA STATE UNIV TEMPE GROUP FOR COMPUTER STUDIES OF STRATEGIES N V FINDLER JUL 83 AFOSR-TR-82-0464 1/1

UNCLASSIFIED AFOSR-82-0340 F/G 12/1 NL

END

MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

4

AD-A133763

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER AFOSR-TR- 82-0464	2. GOVT ACCESSION NO. A133763	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) "AN OVERVIEW OF THE QUASI-OPTIMIZER SYSTEM"		5. TYPE OF REPORT & PERIOD COVERED TECHNICAL
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Nicholas V. Findler		8. CONTRACT OR GRANT NUMBER(s) AFOSR-82-0340
9. PERFORMING ORGANIZATION NAME AND ADDRESS Arizona State University Computer Science Department Tempe, AZ 85287		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS PE61102F; 2304/A2
11. CONTROLLING OFFICE NAME AND ADDRESS AFOSR/NM Building 410 Bolling AFB DC 20332		12. REPORT DATE July 1983
		13. NUMBER OF PAGES 16
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

An overview is given about the Quasi-Optimizer system which observes and measures the behavior of competitive strategies, infers their reasoning, and constructs a descriptive model of each. By evaluating the effectiveness of the components of these strategies and selecting the most satisfactory ones, it generates a normative model which is optimum in the statistical sense.

DD FORM 1 JAN 73 1473

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

AFOSR-TR- 82-0464

AN OVERVIEW OF THE QUASI-OPTIMIZER SYSTEM

Nicholas V. Findler

Group for Computer Studies of Strategies

Computer Science Department

Arizona State University

Tempe, AZ 85287

ABSTRACT

An overview is given about the Quasi-Optimizer system which observes and measures the behavior of competitive strategies, infers their reasoning, and constructs a descriptive model of each. By evaluating the effectiveness of the components of these strategies and selecting the most satisfactory ones, it generates a normative model which is optimum in the statistical sense.

AFOSR-82-0340

83 1[illegible] 252

INTRODUCTION

The Quasi-Optimizer (QO) system [1,2] is a long-term project of which most components have been completed and are currently integrated in a large system. We first discuss its general context. Let us consider an environment in which several organizations compete to achieve some identical goal. (We may assume, for the sake of generality, that a goal vector is specified whose components need not be orthogonal in real-life situations. In business management, for example, the relative share of the market and the volume of sales may be non-orthogonal goal dimensions.) Each organization perceives the environment by observing and measuring certain variables (numeric or symbolic) it considers relevant. Part of the strategy of the organizations aims at interpreting the measurements determining a course of action leading to goal achievement and preventing the adversary from achieving it. At any moment, the "rules" of competition, and the past and current actions of the competitors determine the next state of the environment.

The picture of the environment as perceived by an adversary is unclear because some information may be unavailable, missing (risky or uncertain -- according to whether or not the relevant a priori probability distributions are known, respectively) or may be obscured by noise. Noise may be caused by latent environmental factors or deliberate obfuscation by the competitors. There may also be conflicts and biases within an organization (e.g., rivalry between different divisions or

personalities), which can perturb its measurements and distort its image of the environment. If a competitor's decisions based on such incomplete or faulty information are less sound than those of others, resources will be wasted and goal attainment will be further removed.

If a new organization wants to enter such a confrontation, it must develop a strategy for itself. Assume that this strategy is to incorporate the best components of the extant adversaries' strategies. (An extension of this concept is discussed later.) The process must start with a period of <u>passive</u> or <u>active observation</u>, i.e., before or after having entered the confrontation. In this phase, the new organization, therefore, has to construct first a model (usually referred to as a <u>descriptive theory</u>) of every other participant. To select the most satisfactory components of the model strategies, it would assign to each component some measure of quality, i.e, an outcome-dependent <u>credit assignment</u> must be made. (This assumes that the models are of uniform structure such as decision trees or production systems. Furthermore, credit must be assigned not on the basis of immediate outcome but often in relying on long-term considerations because of planning and interacting learning processes in the strategies.)

Both short-term and long-term objectives can be discerned in the behavior of the adversaries. Short-term objectives comprise local and momentary goals, such as to mislead temporarily the others or eliminate one of their resources, but short-term

objectives naturally contribute to the long-term ones. The long-term objectives are achieved through the overall strategy which is an aggregate of the tactics directed toward some short-term objective. A strategy is also more than that. It includes the means for evaluating the adversaries' situation and actions, scheduling of one's own tactics, and making use of feedback from the environment in modifying the rules of tactics both in terms of their contents and their inter-relations. In short, strategy gives tactics its mission and seeks to reap its results.

The strategy obtainable from the best components of the descriptive model strategies is a normative model which is potentially the best of all available ones, on the basis of the information accessible by the new organization. This normative model strategy is in fact only quasi-optimum for four reasons. First, the resulting strategy is optimum only against the original set of strategies considered. Another set may well employ controllers and indicators for decision-making that are superior to any of the "training" set. Second, the strategy is normative only in the statistical sense. Fluctuations in the adversary strategies, whether accidental or deliberate, impair the performance of the QO strategy. Third, the adversary strategies may change over time and some aspects of their dynamic behavior may necessitate a change in the QO strategy. Finally, the generation of both descriptive models and of the normative

model (the QO strategy) is based on approximate and fallible measurements [6].

2. SYSTEM COMPONENTS

As the previous description suggests, QO is a very large system. It was necessary, both for conceptual and technical reasons, to divide it into fairly self-contained components. The rest of the paper briefly discusses these.

2.1 The QO-1 subsystem [3] constructs a descriptive model of static (non-learning) strategies in the form of a decision-tree (see Fig. 1). The user first inputs the total set of decision variables (variables capable of characterizing situations) and the ranges of their possible values. The decision variables may be

.numerically oriented (that is, they assume a number as a value),

.rank numbers,

.symbolic (attributes, ordered or unordered categories),

.structured data (hierarchies, relationships or priorities).

Experience has shown that the total ranges can be mapped (and normalized) onto a numerical scale (0, 128).

FIGURE 1 ABOUT HERE

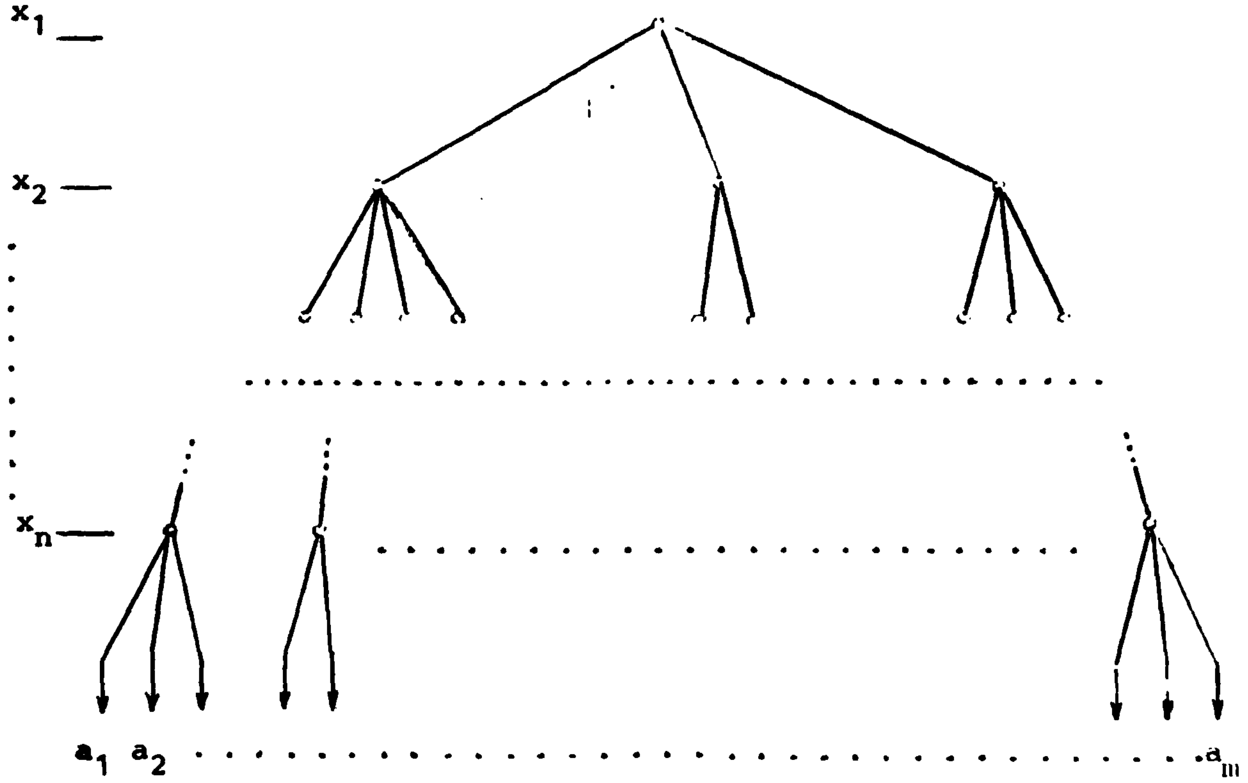

FIGURE 1

In the process of constructing the decision tree, the system discovers which decision variables are relevant for each strategy, that is which are causally connected with the decisions leading to actions.

The user can select for QO-1 one of two modes of operation. In the first, QO-1 assumes the role of a passive observer and records situations and actions taken by the strategy in them, as they happen to occur in an indefinitely long sequence of confrontations. In the second mode, under "laboratory conditions", QO-1 generates situations according to a pre-arranged design and presents them for action to the strategy being modelled. The second mode of operation is, in general, less wasteful but still can be rather expensive when the number of decision variables found relevant for the strategy is large. In the laboratory mode, the user also chooses one of two types of experimental designs, the exhaustive or the "binary chopping" type. In the <u>exhaustive experimentation</u>, the user specifies the <u>maximum meaningful resolution</u> for each decision variable, $\underline{x_i}$, being the smallest observable difference between two distinct adjacent values, $\underline{\Delta x_i}$. The <u>cardinality</u> of $\underline{x_i}$ is the ratio between its range, $\underline{r_i}$ and $\underline{\Delta x_i}$

$$c_i = \frac{r_i}{\Delta x_i}$$

i.e. the maximum number of different values that the decision variable may assume in a sequence of experiments. (The same idea applies to non-numerical variables. These are also mapped onto a

number scale and therefore, $\underline{\Delta x_i}$ is of some computational use.) When QO-1 is operating in the exhaustive mode of experimentation, the total number of responses asked of the strategy is the product of the cardinalities of every decision variable,

$$\prod_{(i)} c_i$$

In the <u>binary chopping mode</u>, QO-1 assumes the strategy response surface to be a weakly monotonic function of every decision variable. The middle value of a decision variable is selected for the next experiment as long as the values returned by the strategy at the two ends of a subrange under study differ by more than a threshold value, $\underline{\Delta R}$, the desired level of precision prespecified by the user. (We ignore in this explanation the redundancy required due to the stochastic nature of the environment.)

Another assumption is implicit in QO-1. Either the strategy response over the whole domain is unidimensional or, if multi-dimensional, the different dimensions of the response do not co-occur in any subrange of the situation space. Therefore, all responses can be mapped onto a unidimensional scale. (We are currently working on removing this restriction.)

An important inductive discovery process can be invoked by the user of QO-1. The system will correlate a stochastic phenomemon/event with situation subranges, if possible. Every time the event occurs, the system computes the subranges within which the values of every decision variable fall with greater

than a given probability. The user is, of course, interested in those decision variables for which the length of the subrange found is relatively small. That indicates a significant causal relationship between that variable and the event in question. (See Fig. 2.)

FIGURE 2 ABOUT HERE

2.2 The QO-2 subsytem [4] extends the power of QO-2. Remember, QO-1 is capable of modelling only static strategies. However, there can be strategies that randomly vary, exhibit some periodic behavior, or learn form experience till they reach a level that is optimum within the limitations of their design. QO-2 is presented a finite sequence of decision trees, snapshots of an evolving strategy made by QO-1 at times when the learning mechanism is "turned off". QO-2 responds by either computing the asymptotic form of the sequence, or requests more snapshots if the input data are "promising" so far but insufficient for the computations at the desired level of statistical significance, or it cannot discover any evidence for a convergent learning process.

The decision tree extrapolated by the QO-2 is then used for generating the normative model.

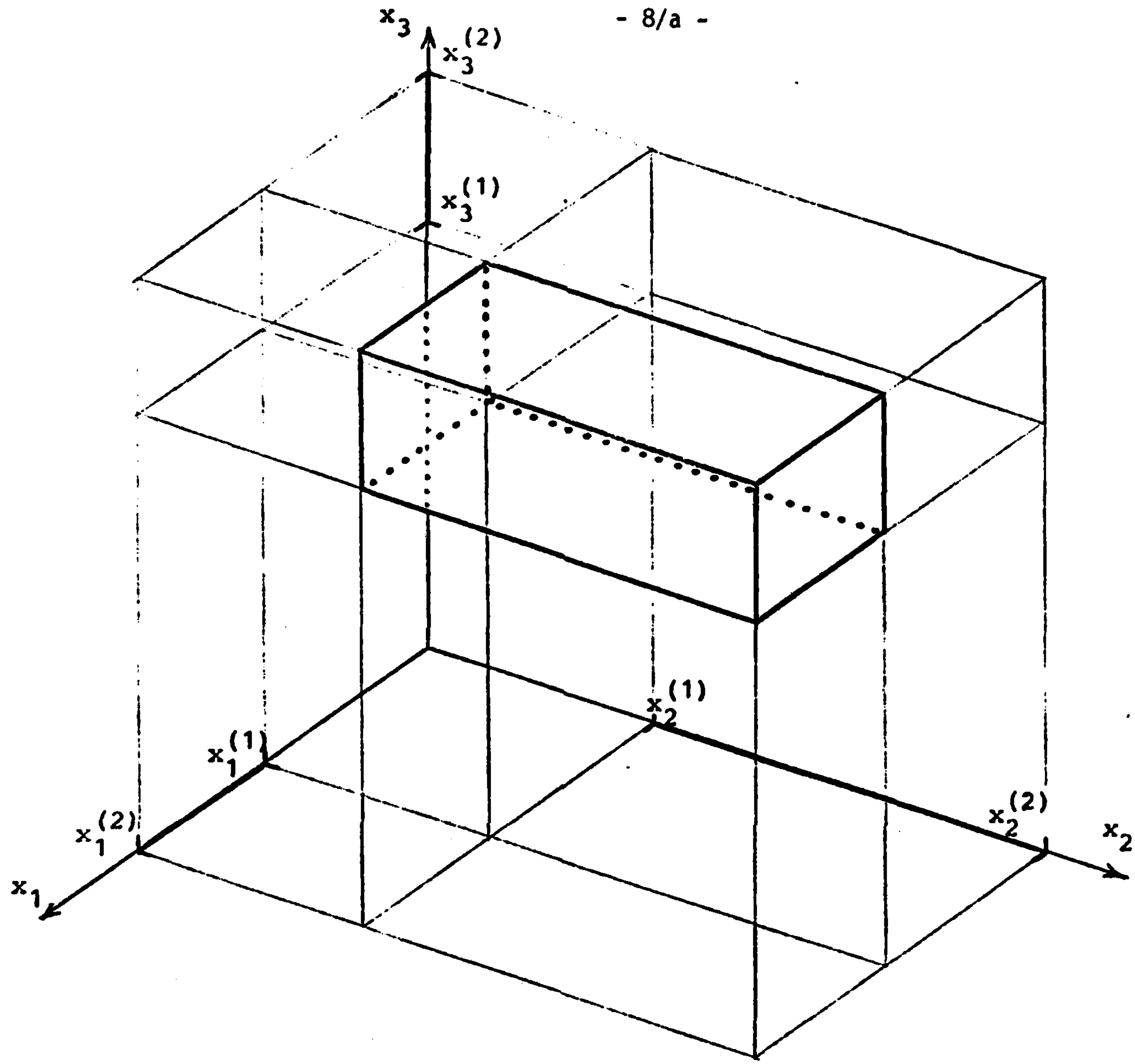

FIGURE 2

2.3 The QO-3 subsystem [5] also enhances the capabilities of QO-1. In addition to the pre-arranged exhaustive and binary chopping modes of experimental design, QO-3 introduces a dynamically evolving design technique. It minimizes the total number of experiments QO-1 has to perform in attaining a prescribed level of precision. QO-3 aims at maintaining a uniform level of sensitivity in the response surface over the whole domain of the decision variable space. Put in simple terms, more experiments must be performed over those regions of the decision variables where the response level changes faster. Ideally, the response levels between two adjacent experimental points should differ by a constant. As an extreme case, when the response level is a linear function of the decision variables (a hyperplane), the balanced incomplete block design satisfies the above requirement. The experiments in fact start with the balanced incomplete block design and make refinements in the grid size whenever the change in the response level warrants it. Furthermore, unlike QO-1, QO-3 does not assume the strategy response to be weakly monotonic.

2.4 The QO-4 subsystem [7] performs the credit assignment, a classical outstanding problem of Artificial Intelligence. It has the following objectives:

(i) To identify and distinguish the components of a strategy;

(ii) To associate with these components good and poor

outcomes of a sequence of actions prescribed by the strategy.

The above indirect "definition" does not concern itself with what a strategy component is, or how one measures "good" and "poor" outcomes.

Let a strategy, S, be described, in accordance with our practice, by a decision tree (DT). (Note that we are not restricted to dealing with static strategies in view of our results in QO-2.)

The environment in which the confrontation takes place is described by the situation vector, $\vec{s}(x_1,\ldots,x_n)$. Its components are the decision variables (which may include measures of the relevant aspects of the history of the confrontation up to that point). The actions prescribed by the stategy, $a_1,\ldots,a_m$, are attached to the leaf level. The same action a_i may appear at several different leaves (see Fig. 3). One could say that the strategy maps a certain number of situations into the same action,

$$S(s_j) \Rightarrow a_i$$
$$S(s_k) \Rightarrow a_i$$
$$\vdots$$

FIGURE 3 ABOUT HERE

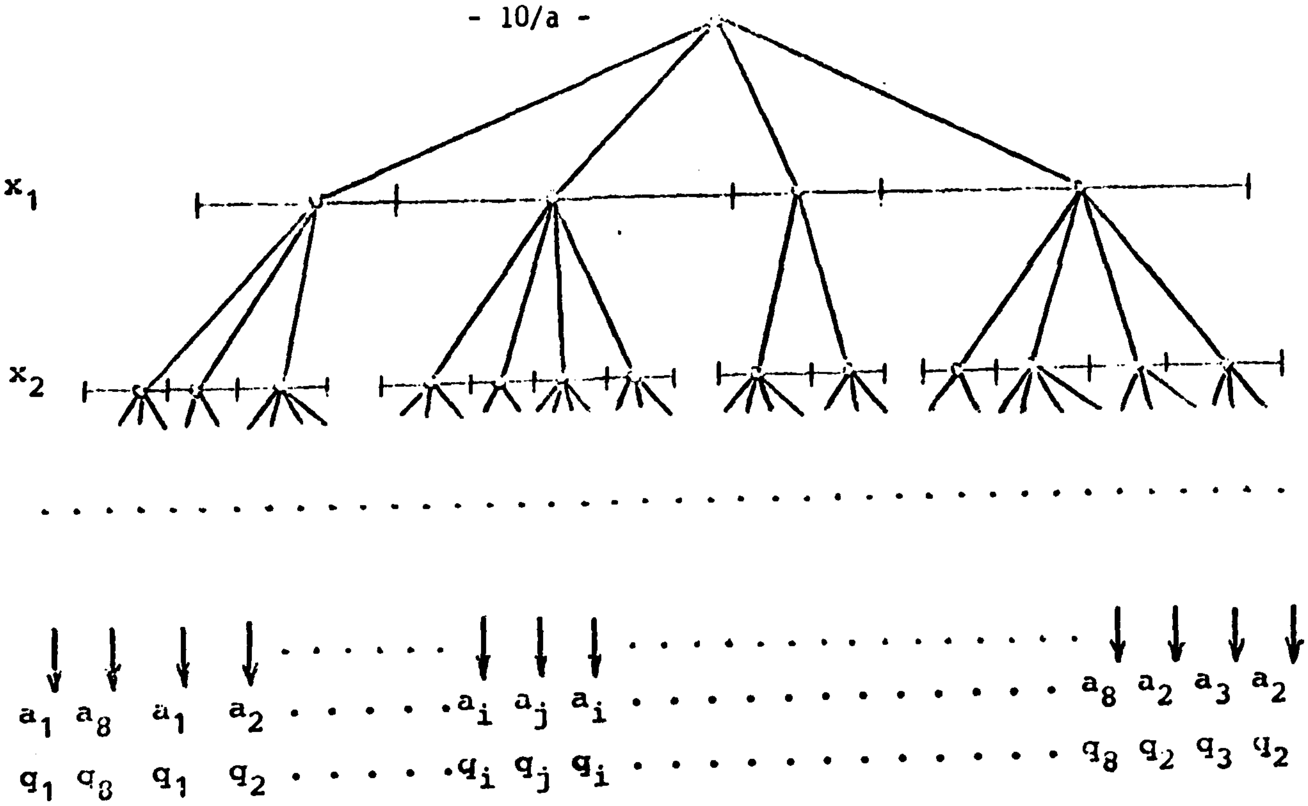

FIGURE 3

Let us now assume that we can establish a measure of the quality of the consequences of every action in the situation,

$$q(\vec{s}, a_i)$$

This measure is in the first approximation independent of the strategy. Further refinements would also consider, for example, long range plans in the strategy at hand, and evaluate the actions not only in terms of immediate outcomes.

Let there be a quality scale, ranging in values between, say, 0 and 100. (Remember the current assumption about one-dimension strategy responses.) Let us define two sliding boundary points on this scale, B and G. (Their location may change as a result of the learning process described.) We shall call the quality of an action 'bad' if the corresponding value is between 0 and B, and 'good' if it is between G and 100. (See Fig. 4.)

FIGURE 4 ABOUT HERE

We are now coming to an operational definition of a strategy component. We have noted before that a certain action, a_i, may be prescribed by the strategy in a number of different situations. Let all the pathways, in the decision tree representing the strategy, leading to a_i form the class

$$U_i : \{u_j, u_k, \ldots\}$$

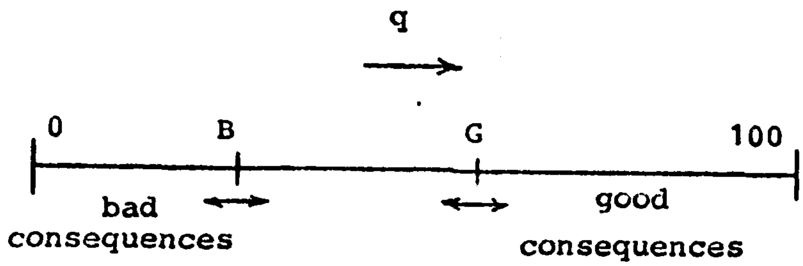

FIGURE 4

Here $\underline{u_j}$, is the pathway that corresponds to the situation vector $\underline{\vec{s}_j}$. There are two important subclasses, $\underline{U_i}^{(b)}$ and $\underline{U_i}^{(g)}$, of the class $\underline{U_i}$ which contains the pathways to $\underline{a_i}$, producing bad and good consequences, respectively. Let them be

$$U_i^{(b)} : \{u_m, u_n, \ldots\}$$
$$U_i^{(g)} : \{u_s, u_t, \ldots\}$$

A <u>strategy component</u> is defined as the set of characteristic features <u>contrasting</u> $\underline{U_i}^{(b)}$ and $\underline{U_i}^{(g)}$. The <u>characteristic feature</u> of a pahway is the Boolean AND (in the general case, also OR and NOT) of its atomic properties. Finally, an <u>atomic property</u> of a pathway is the subrange of a decision variable value through which the pathway goes, at any level between the root and the leaves.

The algorithm first forms <u>all</u> the characteristic features of the first two subclasses of pathways $\underline{U_i}^{(b)}$ and $\underline{U_i}^{(g)}$. It then discards all but the <u>discriminating</u> features. We are planning a high-level learning process that will maximize the <u>power of discrimination</u> between the two subclasses. The location of the boundary points <u>B</u> and <u>G</u> will be systematically changed and eventually optimized on the quality scale so that the sum of the probabilities of making Type I and Type II is miniumum. These error types are analogous to Type I and Type II errors in statistical hypothesis testing, and refer to <u>accepting a wrong pathway</u> in and <u>rejecting a correct pathway from a subclass of pathways</u>, respectively.

We also study different types of uncertainties and sources of noise, all of which could lead to errors in the credit assignment task. These are as follows:

.a situation may not be described exactly because of measurement errors or latent variables;

.an action prescribed by the strategy in a given situation may not be executed exactly;

.a certain action may not always have the same consequence in the same situation because the environment may at times change rapidly and also for the reasons listed above.

2.5 The QO-5 subsystem constructs a 'Super Strategy'. Our design for the QO-5 is based on the concepts defined in connection with QO-4. Strategy components from different strategies leading to the same action are ranked in the order of the quality of consequences. The best components are chosen so that the whole decision variable space is covered (responded to) by the set of strategy components to form the Super Strategy. (These "best" components may be further improved when the QO strategy is employed in confronting other strategies -- a method called 'experientialization'.)

2.6 The QO-6 subsystem has the objective of eliminating the redundancies and inconsistencies of the Super Strategy while maintaining its completeness and soundness. The techniques being considered and used range from theorem proving, at the abstract

end of the spectrum, to statistical and heuristic ideas at the constructive end of the spectrum.

3. CURRENT STATUS AND FINAL COMMENTS

At the time of writing this report, subsystems QO-1, QO-2, QO-3, QO-4 and QO-5 have been completed and QO-6 is being implemented. There is some work to be done in integrating these modules before we can make practical use of them.

We feel that we are producing a fairly general system that, besides having theoretical interest in the study of strategies, may prove useful in complex optimization problems. Furthermore, ideas embedded in the project, such as automatic generation of computer models, dynamically evolving design of experiments, and feature extraction-oriented credit assignment, can be of value to Artificial Intelligence research in general.

4. ACKNOWLEDGEMENTS

I am indebted to the members of the Group for Computer Studies of Strategies -- in particular to George L. Sicherman, Bede McCall, Neal Mazur, Robert Cromp, Timothy Bickmore, Michael Belofsky, Chung-Shu Chang and Paul Duerig. Our work has been supported by AFOSR Grants 81-0220 and 82-0340.

5. REFERENCES

[1] Findler, N. V. and J. van Leeuwen: The complexity of decision trees, the Quasi-Optimizer, and the power of heuristic rules (Information and Control, 40 pp. 1-19, 1979).

[2] Findler, N. V.: On a computer based theory of strategies (To appear in Kybernetes).

[3] Findler, N. V. and J. P. Martins: On automating computer model construction -- The second step toward a Quasi-Optimizer system (Journal on Information and Optimization Sciences, 2, pp. 119-136, 1980).

[4] Findler, N. V., N. Mazur and B. McCall: A note on computing the asymptotic form of a limited sequence of decision trees (To appear in Information Sciences).

[5] Findler, N. V.: On a system that dynamically generates its own experimental design (COMSTAT82, Fifth Internat. Symposium on Computational Statistics, Toulouse, France. Physica Verlag: Würzburg, West Germany, 1982).

[6] Findler, N. V: On automatic generation of descriptive and normative theories (Invited paper, IEEE Internat. Conf. on Systems, Man and Cybernetics, India, 1983).

[7] Findler, N. V. and B. B. McCall: A conceptual framework and a heuristic program for the credit assignment problem (Submitted for publication).

LEGEND FOR THE FIGURES

FIGURE 1 -- Schematic Representation of a Static Decision Tree. Each level of the tree is identified with one of the decision variables x_1, x_2, ... x_h. The leaves attached to the branches at the last level, a_1, a_2,... represent actions. A path down from the root to an action in the decision tree is defined by a particular combination of values of the decision variables and characterizes the environment as perceived by the strategy which is represented by the decision tree.

FIGURE 2 -- The Result of an Inductive Discovery Process. A stochastic event is correlated with a region of the situation vector points into the polyhedron shown (defined by the subranges of the decision variables x_1, x_2, ...) 95% of the times when a certain event occurs.

FIGURE 3 -- Supportive Diagram for the Credit Assignment Problem. A schematic decision tree. The actions are attached at the leaf level. The quality measures of consequences of each action are also indicated.

FIGURE 4 -- The Scale of Quality of Consequences. Common features of pathways on the decision tree, discriminating between bad and good consequences, help in defining strategy components.

END

FILMED

11-83

DTIC